Vlasov Simulation Methods for plasmas and beams I

Eric Sonnendrücker
IRMA, Université Louis Pasteur
Strasbourg, France
Co-workers: F. Filbet, M. Gutnic, I. Paun,
N. Besse, J.-L. Lemaire (CEA)

Berkeley, July 24, 2002

Outline

- Why use direct Vlasov methods ?
- Overview of Vlasov methods for plasma and beam simulations
  - ⋆ Particle methods
  - ⋆ Spectral methods
  - ⋆ Eulerian methods.
- Applications: beam propagation in uniform and periodic focusing channels.

Why use direct Vlasov methods ?

- Important noise in PIC methods especially in poorly populated regions of phase space makes it hard to see phenomena like e.g.
  - ⋆ particle trapping (strong Landau damping) in plasmas
  - ⋆ halo formation in beams
- Computers now powerful enough to do realistic physics using a grid in phase space.
- Provides alternative to PIC for code benchmarking.

The Vlasov equation

The distribution function $f(x, v, t)$ is solution of

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + \frac{q}{m} (E + v \times B) \cdot \nabla_v f = 0,$$

generally coupled with the Poisson or Maxwell equations.

- Numerical challenges:

  ⋆ Six-dimensional space

  ⋆ appearance of very small scales

Overview of Numerical Methods for the Vlasov equation

Different classes of methods:

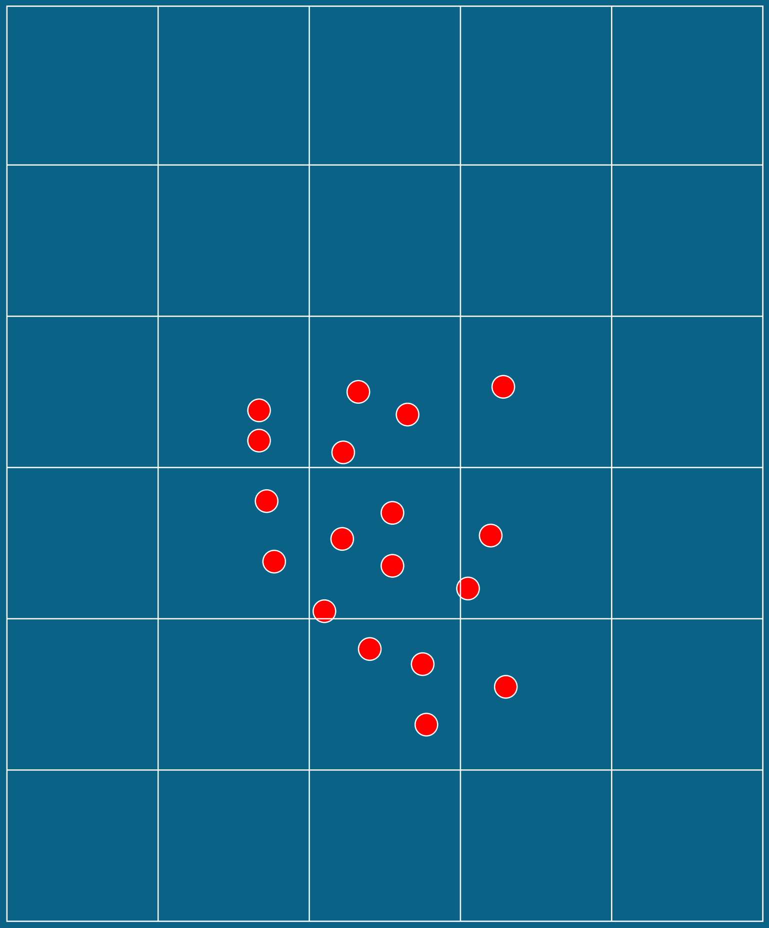
- Particle Methods
- Spectral methods
- Methods based on a grid of phase-space

Particle Methods

- Particle-In-Cell (PIC) method :
  - ⋆ Idea: Follow particle trajectories, use grid field solve.
  - ⋆ Literature: *Birdsall-Langdon, Hockney-Eastwood, Neunzert-Wick, Cottet-Raviart, Victory-Allen*
- SPH type methods:
  - ⋆ Idea: Compute interaction between finite sized macro-particles.
  - ⋆ Literature: *Bateson-Hewett, …*

The Particle-In-Cell Method

- Particles advanced using characteristics
- Fields are computed on grid
- Particle data scattered to surrounding grid points to compute charge and current densities
- Fields are computed on particles using interpolation



Overview of Numerical Methods for the Vlasov equation

Different classes of methods:

- Particle Methods
- Spectral methods
- Methods based on a grid of phase-space

Spectral Methods

- Fourier-(Fourier) methods: *Knorr, Klimas-Farell*

  ⋆ Split between position and velocity advection

  $$\frac{\partial f}{\partial t} + v \cdot \nabla_x f \; = \; 0, \qquad \frac{\partial f}{\partial t} + \frac{q}{m} E \cdot \nabla_v f \; = \; 0.$$

  ⋆ Perform Fourier transform to be able to compute exactly the phase advance for each mode, e. g. $\hat{f}_k^*(v) = \hat{f}_k^n(v) \, \exp(-i \, 2 \, \pi \, k \cdot v \, \Delta t / 2 \, L)$.

- (Fourier)-Hermite methods : *Shoucri, Holloway, Pulvirenti-Wick*

Overview of Numerical Methods for the Vlasov equation

Different classes of methods:

- Particle Methods
- Spectral methods
- Methods based on a grid of phase-space

Eulerian Methods

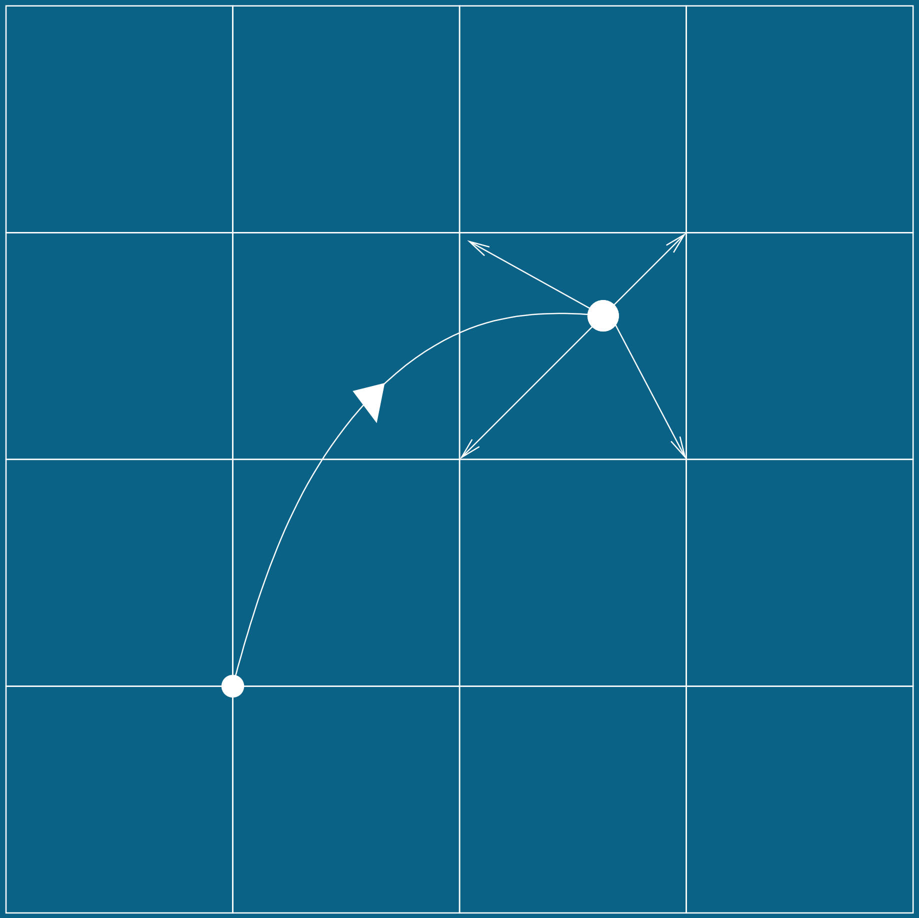
- Forward semi-Lagrangian (PIC with reconstruction): *Denavit*
- Backward semi-Lagrangian: *Cheng-Knorr, ES-Roche-Bertrand-Ghizzo, , Nakamura-Yabe, ...*
- Finite Volume: *Boris-Book, Fijalkow, Filbet-ES-Bertrand*
- Energy conserving Finite Difference Method: *Filbet-Kazantsev-ES*

The forward semi-Lagrangian Method

- f conserved along characteristics

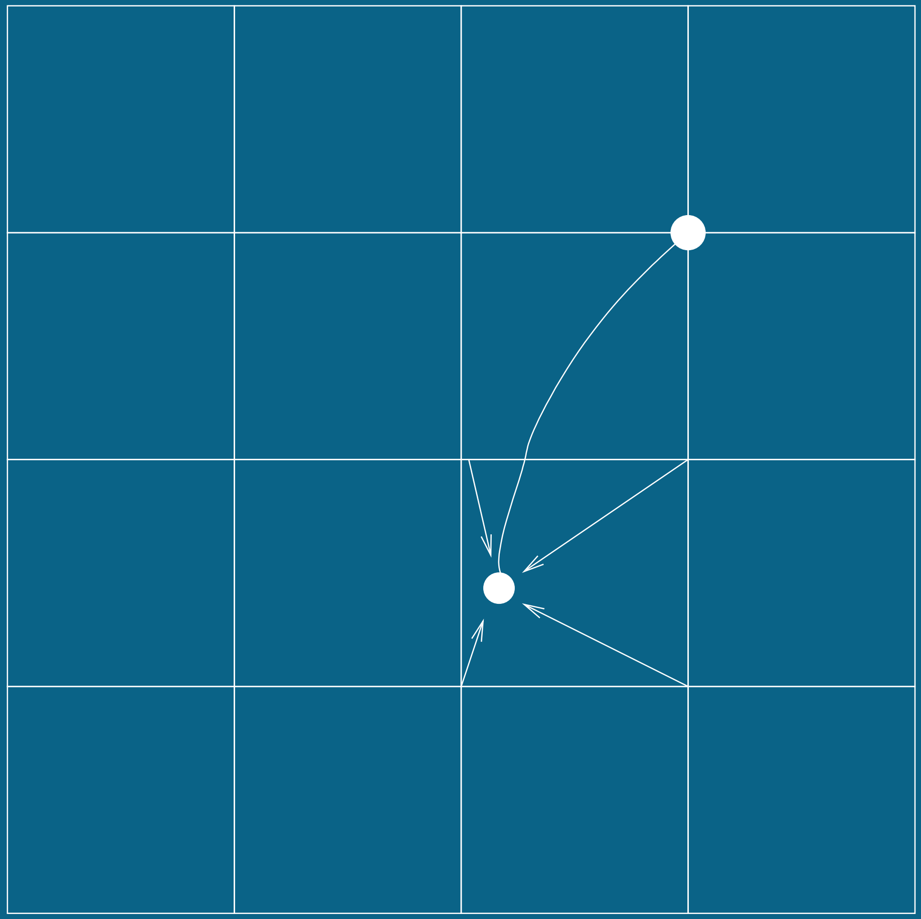
$$\frac{dX}{dt} = V, \qquad \frac{dV}{dt} = E(X,t)$$

- Grid points advanced along characteristics
- Values of f scattered to grid



The backward semi-Lagrangian Method

- f conserved along characteristics
- Find the origin of the characteristics ending at the grid points
- Interpolate old value at origin of characteristics from known grid values → High order interpolation needed

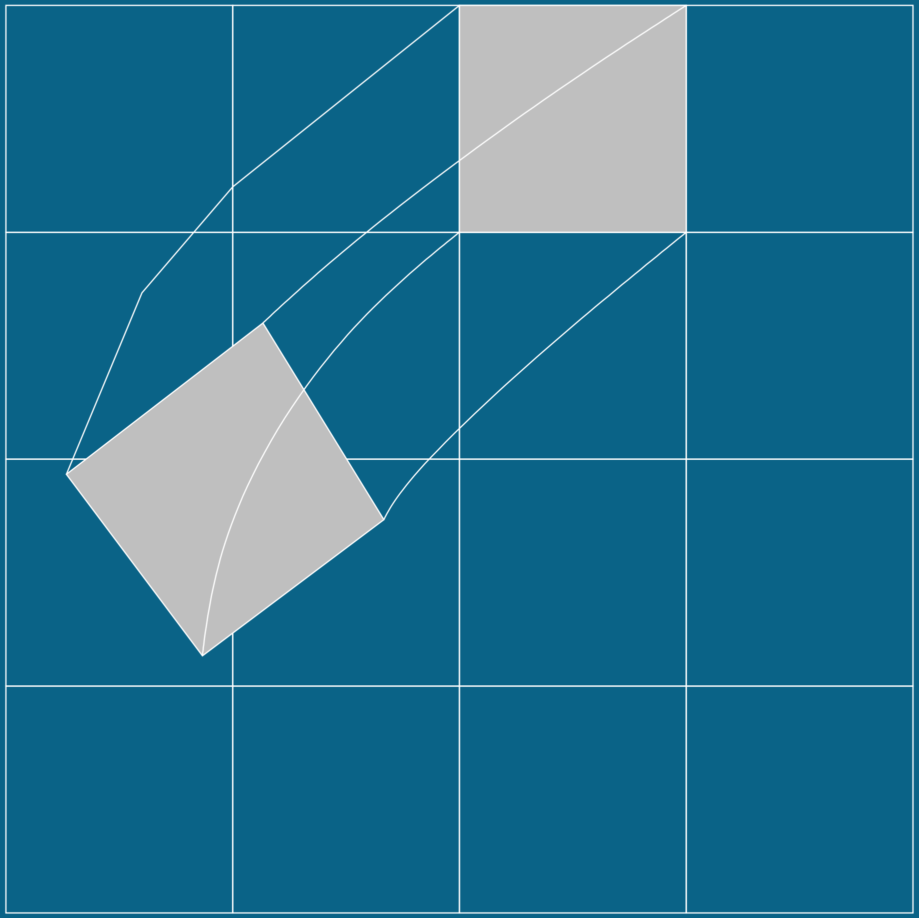


Interpolation schemes

- Lagrange: polynomial interpolation using node values
- Hermite: polynomial interpolation using values of functions and derivatives. Values of derivatives can be computed numerically or by advecting gradients (CIP).
- Spline: cubic spline in original method by Cheng and Knorr.

The Finite Volume Method

- $\int_V f\,dx\,dv$ conserved along characteristics
- Three steps:
  - ⋆ Reconstruction
  - ⋆ Resolution
  - ⋆ Projection
- Slope limiters for positivity



Description of the FV method

Consider the transport equation in conservative form

$$\partial_t f + \partial_x \left(\mathbf{u}(t,x)\, f \right) = 0, \quad \forall (t,x), \tag{1}$$

and define the characteristic curves:

$$\frac{d\,\mathrm{x}}{ds}(s) = \mathbf{u}(s, \mathrm{x}(s)), \quad \mathrm{x}(t) = x. \tag{2}$$

We denote by x(s,t,x) the solution of (2), then

$$\forall K \subset \mathbb{R}, \quad \int_K f(t,x)dx = \int_{\mathrm{x}(s,t,K)} f(s,x)dx. \tag{3}$$

The numerical scheme

The starting point of the conservative method is to use this last property to compute the approximation on a mesh of the phase space:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f(t^{n+1},x)dx = \int_{\mathrm{x}(t^n,t^{n+1},x_{i-1/2})}^{\mathrm{x}(t^n,t^{n+1},x_{i+1/2})} f(t^n,x)dx,$$

then, we set

$$\Phi_{i+1/2}(t^n) = \int_{\mathrm{x}(t^n,t^{n+1},x_{i+1/2})}^{x_{i+1/2}} f(t^n,x)dx.$$

The method can be split into three steps to go from the time t^n to $t^{n+1} = t^n + \Delta t$:

1. Reconstruction
2. Resolution
3. Projection

The reconstruction step

We use an interpolation method on the cell $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$,

$$f_h(x) = f_i + \frac{\epsilon_i^+}{6\,\Delta x^2}\alpha(x)\,(f_{i+1} - f_i) + \frac{\epsilon_i^-}{6\,\Delta x^2}\beta(x)\,(f_i - f_{i-1}),$$

to reconstruct high order polynomial, where

$$\alpha(x) = 2\,(x - x_i)(x - x_{i-3/2}) + (x - x_{i-1/2})(x - x_{i+1/2}),$$
$$\beta(x) = 2\,(x - x_i)(x - x_{i+3/2}) + (x - x_{i-1/2})(x - x_{i+1/2}),$$

ϵ_i^+ and ϵ_i^- are slope correctors allowing to preserve positivity.

The resolution and projection steps

The resolution step. We approximate the caracteristic curves by solving the system of ODE's (2).

The projection step. We compute the cell-average of the approximation at time t^{n+1}

$$f_i^{n+1} = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} f_h(t^{n+1}, x) dx.$$

Evaluation of the average allows to ignore fine details of the exact solution which may be very costly to compute.

A L^2-norm conserving scheme

- Introduced by Arakawa (1966) for equations of the form

$$\frac{\partial f}{\partial t} + J(\psi, f) = 0,$$

for Vlasov $\psi = \varphi - \frac{v^2}{2} \quad \text{and} \quad J(\psi, f) = \frac{\partial \psi}{\partial x}\frac{\partial f}{\partial v} - \frac{\partial \psi}{\partial v}\frac{\partial f}{\partial x},$

- Second and fourth order implemented:
  - ⋆ Particle conservation: $\frac{d}{dt}\int_{\mathbb{R}^2} f(t)\, dx\, dv = 0.$
  - ⋆ Energy conservation: $\frac{d}{dt}\int_{\mathbb{R}^2} f(t)\psi(t)\, dx\, dv = 0.$
  - ⋆ Conservation of $\|f\|_{L^2}$.

Stabilization of the method

- Strong oscillations in presence of filamentation.
- Collision model of the form $\frac{\partial f}{\partial t} = \frac{\partial \mathcal{J}}{\partial v}$, where $\mathcal{J}$ is chosen so as to maximize entropy and conserve as many moments $\int f v^k \, dv$ as desired ($Sommeria\ and\ Robert$).
- A Lagrange multiplier technique yields (for $k = 2$):

$$\frac{\partial f}{\partial t} + J(\psi, f) = \alpha \frac{\partial}{\partial v}\left(\frac{\partial f}{\partial v} + A_1 f - A_2 f v\right),$$

with $A_1 = \frac{u_0}{\epsilon - u_0^2/n}$, and $A_2 = \frac{n}{\epsilon - u_0^2/n}$.

Properties of interpolation schemes

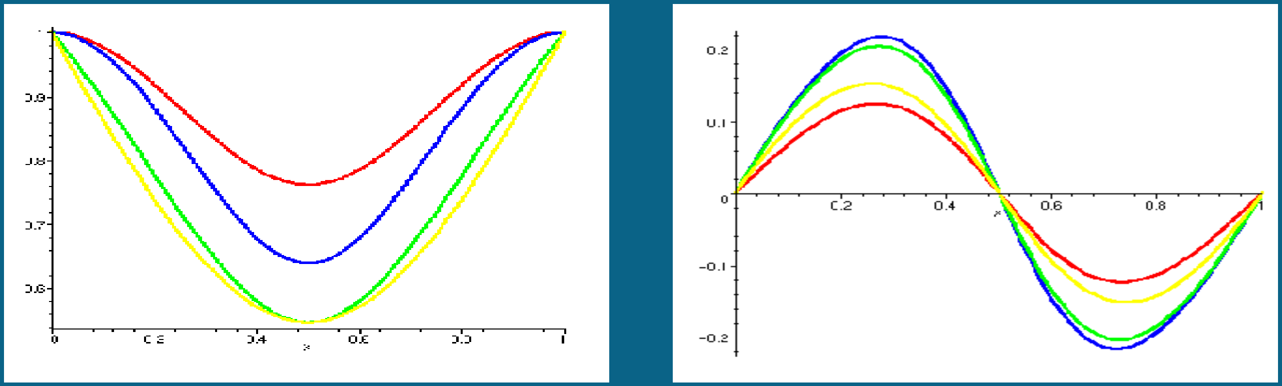


Figure 1: *Amplification factor and phase error with respect to α for a fixed mode k. FBM method (yellow) and third order PFC (green), semi-Lagrangian method with a cubic Hermite (blue) and spline (red) interpolation.*

Evolution of L^1 and L^2 norms

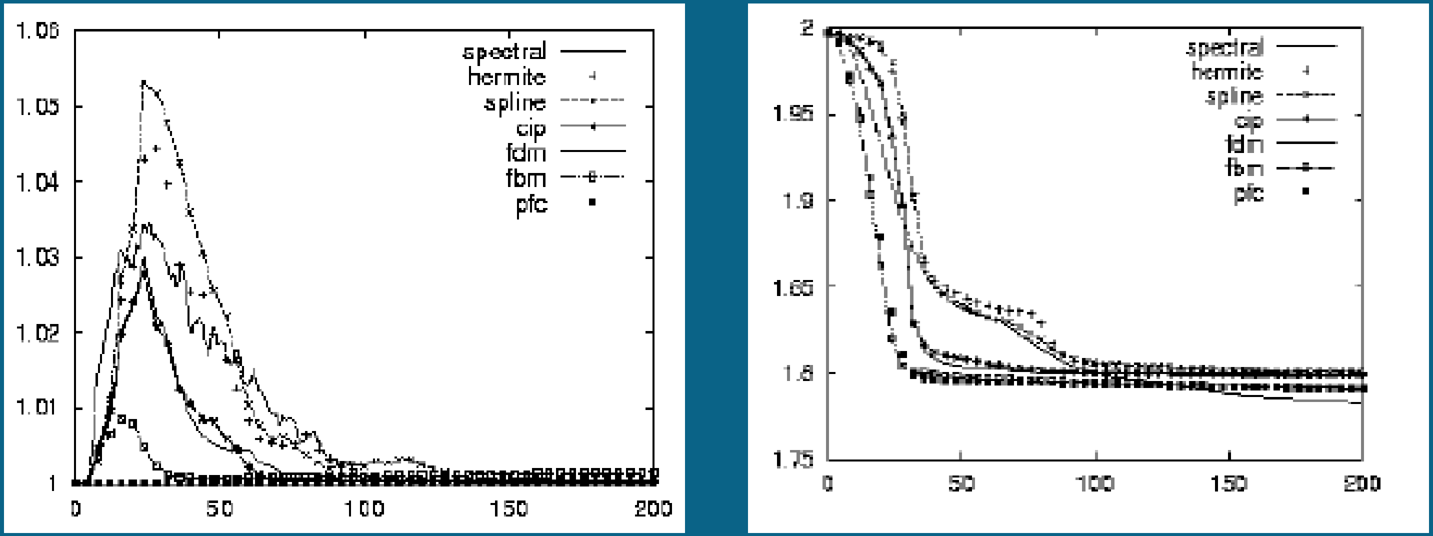


Figure 2: *Time development of numerical* L^1 *and* L^2 *norms for the non linear Landau damping test.*

Evolution of L^2 norm and kinetic entropy

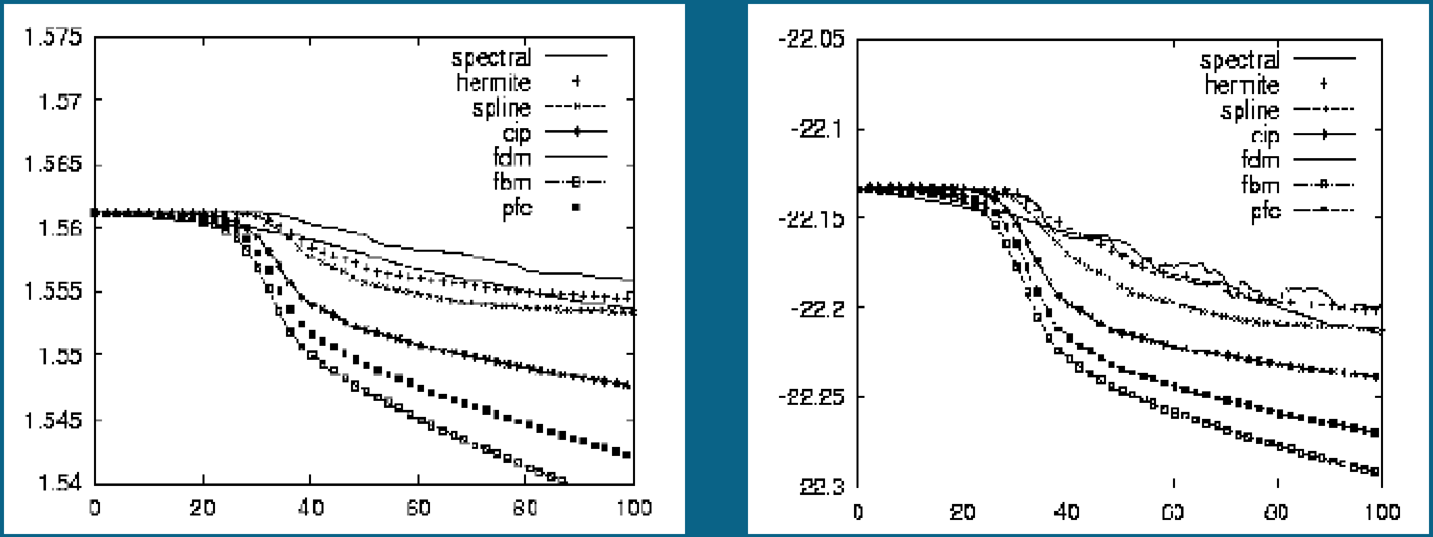
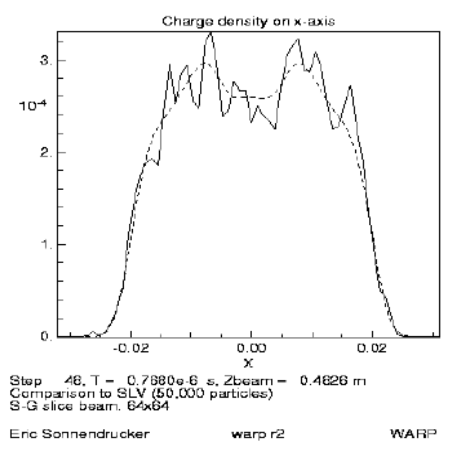
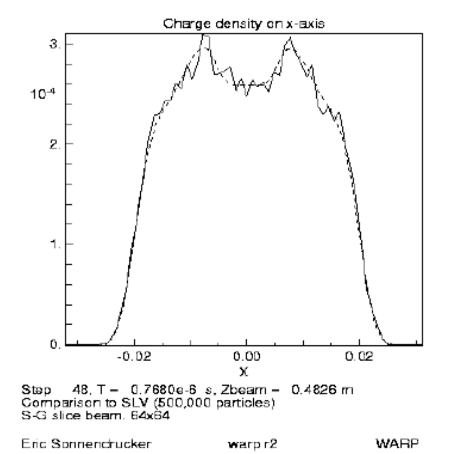
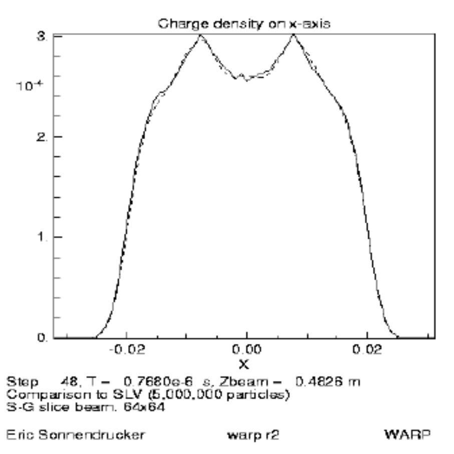


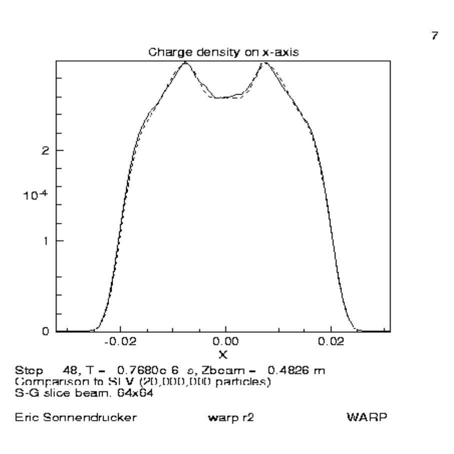
Figure 3: *Time development of numerical L^2 norm and entropy of $f(t)$ for the two stream instability test.*

Comparison of Eulerian and particle methods









Computation time

Numerical method	32 × 32 points	32 × 64 points	32 × 128 points
FBM	03.33 sec.	05.39 sec.	10.80 sec.
PFC	03.56 sec.	06.28 sec.	11.20 sec.
FDM	17.22 sec.	35.27 sec.	71.20 sec.
SPECTRAL	04.10 sec.	08.25 sec.	16.90 sec.
CIP	13.83 sec.	21.40 sec.	43.24 sec.
SL SPLINE	06.12 sec.	10.55 sec.	20.90 sec.
SL HERMITE	03.60 sec.	06.90 sec.	11.00 sec.

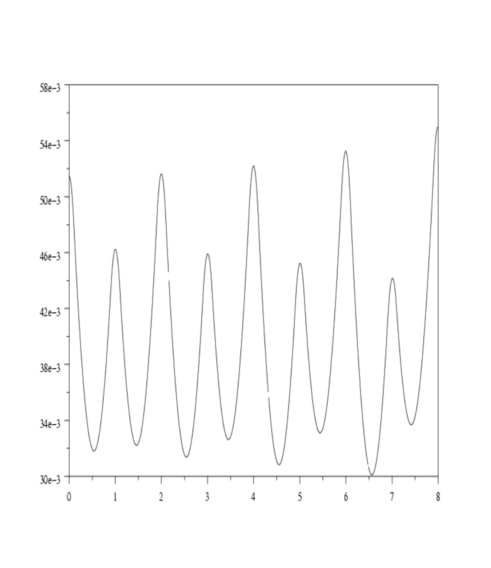
Parallelization

- Transpose of distribution function performed at each split step
  - ⋆ Perfect load balacing at any given time
  - ⋆ Huge data transfer (needs good network)
  - ⋆ Concurrency of numerical computations and communications (E. Violard, F. Filbet)
- Option might be optimal for non local interpolations, probably not for local ones ??

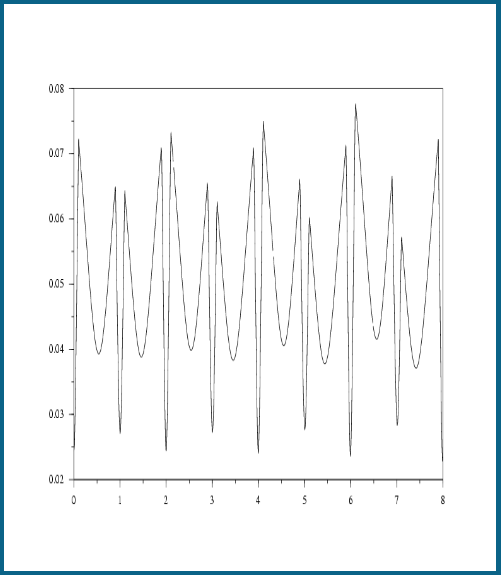
Beam propagation in uniform and periodic focusing channel

- Uniform focusing: Potassium ions, 80 keV, 0.1 A, tune depression 0.25, rms-matched semi-gaussian beam

- Periodic focusing: Same beam data, emittance 0.005, beam radius 0.1 m, lattice period 1 m, focusing length 0.2 m. phase advance: $\sigma_0 = 113$ degrees, $\sigma = 50$ degrees.

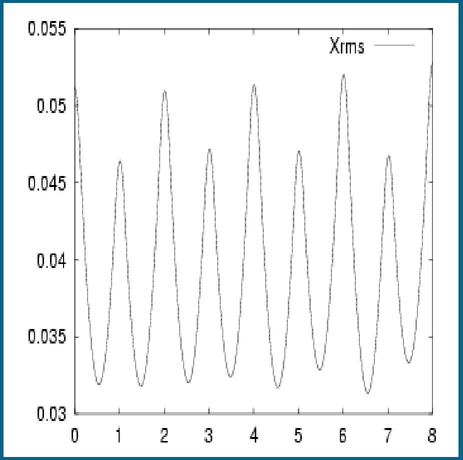


(1)

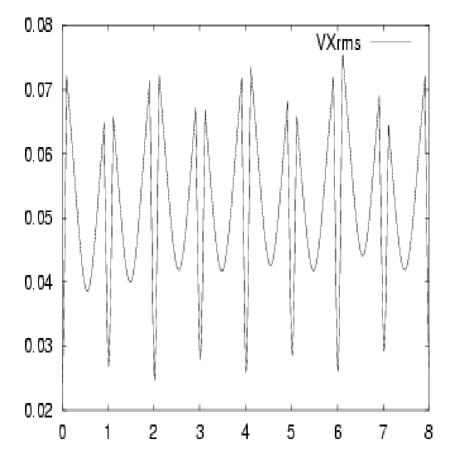


(2)

r_{rms} and r'_{rms} with envelope code.



(3)



(4)

r_{rms} and r'_{rms} with VADOR code.

Conclusions and perspectives

- History
  - ⋆ 1970s: At the beginning of Vlasov simulations when only 1D simulations could be performed, all methods coexisted.
  - ⋆ 1980s-1990s: multi-dimensional simulations could be done only with particle methods.
  - ⋆ End 1990s-2000s: Computer power is now enough for multi-dimensional simulations using non particle methods.

- Future
  - ⋆ Different methods should coexist depending on problem at hand.
  - ⋆ Next week
    - ∗ Unstructured meshes.
    - ∗ Specific handling of axisymmetric beams.
    - ∗ Adaptive meshes: semi-Lagrangian method on adaptive mesh using wavelet techniques to handle refinement.

A few papers and web pages

- ⋆ Nicolas Besse : Semi-Lagrangian schemes for the Vlasov equation on an unstructured mesh of phase space
  http://www-irma.u-strasbg.fr/irma/publications/2002/02028.shtml
- ⋆ Francis Filbet, Jean-Louis Lemaire, Eric Sonnendrucker : Direct axisymmetric Vlasov simulations of space charge dominated beams
  http://www-irma.u-strasbg.fr/irma/publications/2002/02009.shtml
- ⋆ Francis Filbet, Eric Sonnendrucker : Comparison of Eulerian Vlasov Solvers
  http://www-irma.u-strasbg.fr/irma/publications/2001/01035.shtml

- Francis Filbet, Eric Sonnendrucker, Pierre Bertrand: Conservative numerical schemes for the Vlasov equation. J. Comp. Phys. Volume 172, Number 1 pp. 166-188 (2001).
- The VADOR code:
  `http://www-irma.u-strasbg.fr/~filbet`

Preview of next week's talk

- Axisymmetric simulations.
- An adaptive method based on wavelets.

The axisymmetric Vlasov equation

The distribution function $f(r, v_r, v_\theta, t)$ is solution of

$$\frac{\partial f}{\partial t} + v_r \frac{\partial f}{\partial r} + \left(\frac{qE_s}{m} + \frac{q\,B_z}{m} v_\theta + \frac{v_\theta^2}{r} \right) \frac{\partial f}{\partial v_r} - \left(\frac{q\,B_z}{m} v_r + \frac{v_\theta\, v_r}{r} \right) \frac{\partial f}{\partial v_\theta} = 0,$$

where B_z is external and E_s given by Poisson's equation

$$\frac{1}{r}\frac{\partial\, r E_s}{\partial r} = \rho(t, r)/\varepsilon_0, \quad \rho(t, r) = q \int_{\mathbb{R}^2} f(t, r, v_r, v_\theta) dv_r\, dv_\theta.$$

Invariants of the Vlasov equation

In order to reduce the dimension of the problem we use the invariance of the canonical angular momentum

$$P(r, v_\theta) = mrv_\theta + \frac{r^2}{2} qB_z.$$

Denoting by $I = \frac{P}{m}$ and making the change of variable $(r, v_r, v_\theta) \longrightarrow (r, v_r, I)$ with $v_\theta = \frac{I}{r} - \frac{1}{2}\frac{q\,B_z}{m} r$, we get

$$\frac{\partial f}{\partial t} + v_r \frac{\partial f}{\partial r} + \left(\frac{q}{m} E_s(t, r) + \frac{I^2}{r^3} - \frac{1}{4} \left(\frac{q\,B_z}{m} \right)^2 r \right) \frac{\partial f}{\partial v_r} = 0.$$

Discretization of the axisymmetric Vlasov equation

- Invariant I is a parameter but needs careful discretization. Characteric curves of the form

$$\frac{\omega^2}{2} r^2 + v_r^2 + \frac{I^2}{r^2} = const.$$

$\longrightarrow$ necessary to control I/r hence I is discretized according to $I = \omega\, r^2$, in vicinity of axis.

- Difficulty near $r = 0$ because of I^2/R^3 term.

- Time-Splitting scheme:
  - ⋆ Advection in r : $\frac{\partial f^*}{\partial t} + v_r \frac{\partial f^*}{\partial r} = 0,$
  - ⋆ Advection in v_r :

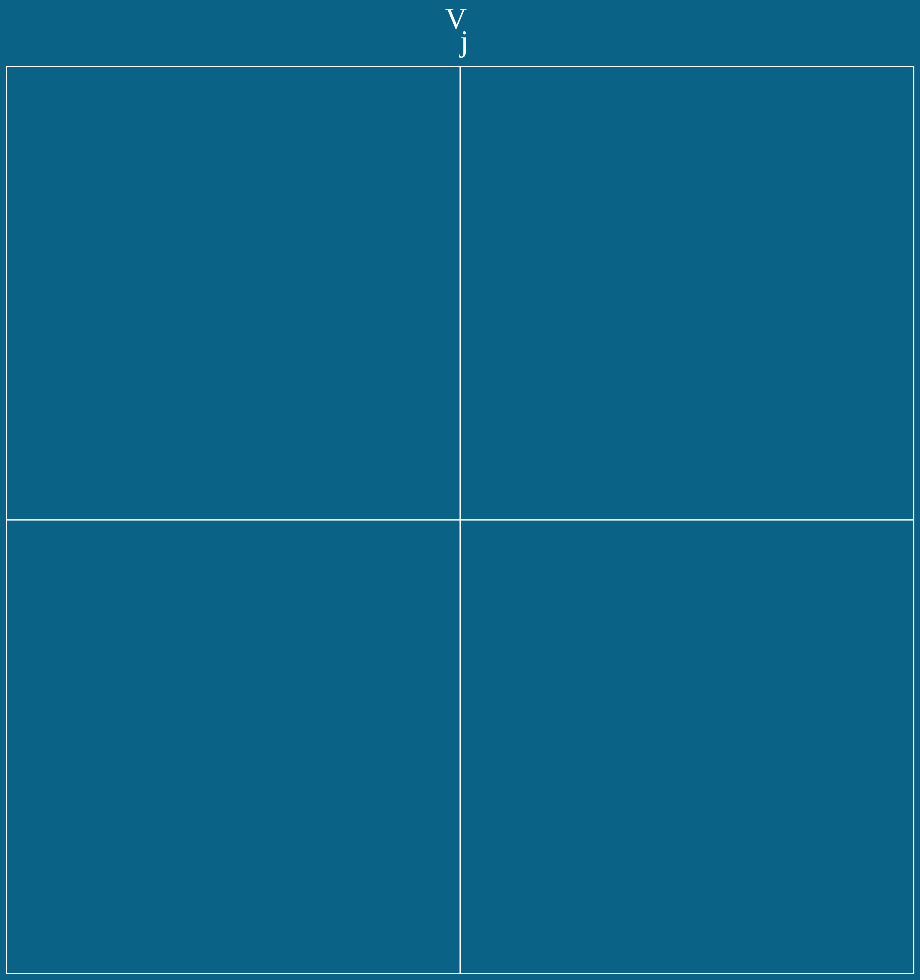
$$\frac{\partial f^{**}}{\partial t} + \left(\frac{q}{m} E_s(t,r) + \frac{I^2}{r^3} - \frac{1}{4} \left(\frac{q\, B_z}{m} \right)^2 r \right) \frac{\partial f^{**}}{\partial v_r} = 0.$$

- Cubic Hermite interpolation with numerical computation of derivatives by a fourth order finite difference scheme

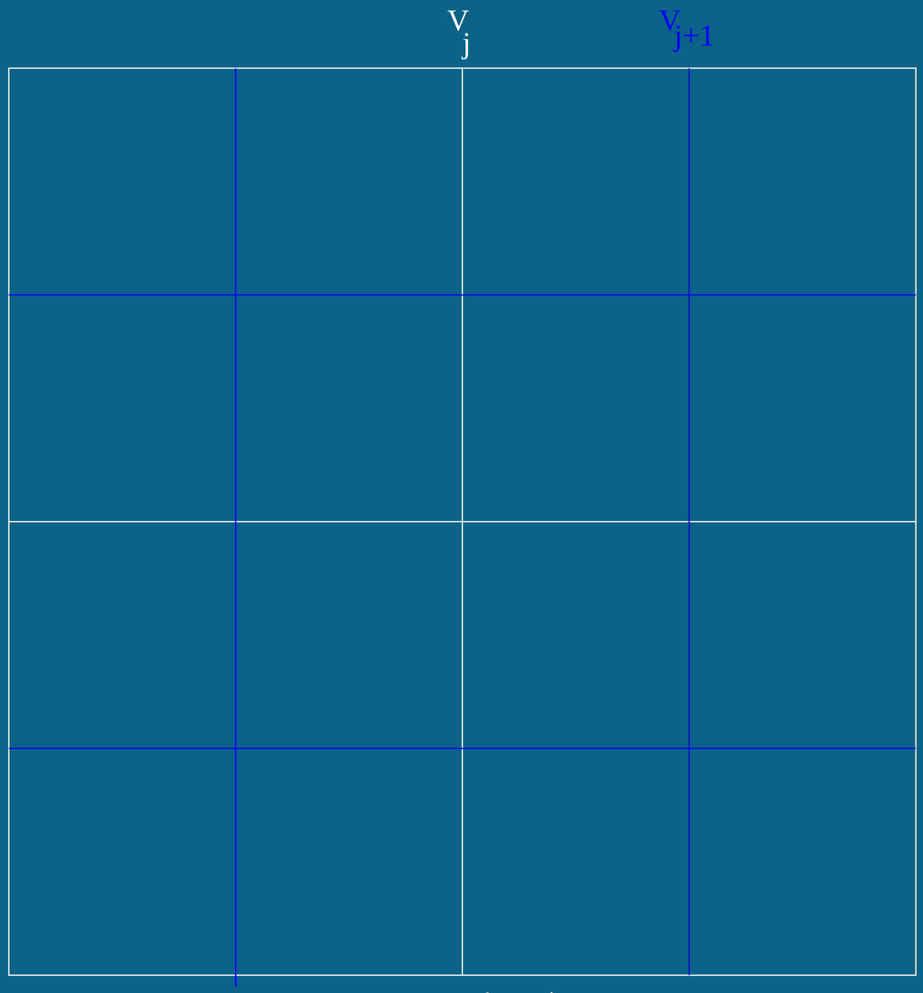
$$\partial_r f_i^n = \frac{1}{12 \Delta r} \left[8 \left[f_{i+1}^n - f_{i-1}^n \right] - \left[f_{i+2}^n - f_{i-2}^n \right] \right].$$

Adaptive semi-Lagrangian method

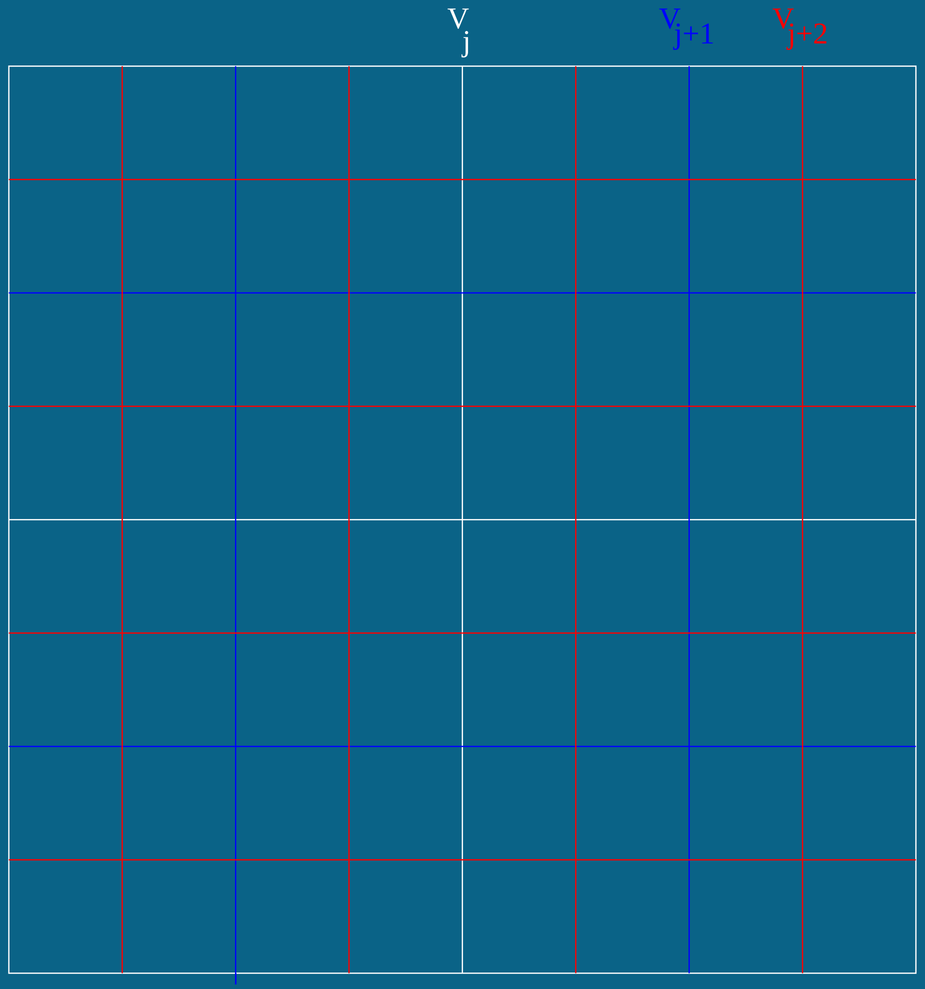
- We want to optimize the number of grid points for a given numerical error.
- Multi-resolution techniques using interpolating wavelets are well suited to determine where refinement is needed.
- Principle of the method
  - ⋆ Use different levels of meshes
  - ⋆ At one given level, decompose gridfunction into gridfunction at coarser level + details.



Grid G_j, grid points $x^j_k = k\,2^j$, level j



Grid G_{j+1}, grid points $x_k^{j+1} = k\,2^{j+1}$, level $j+1$



Grid G_{j+2}, grid points $x_k^{j+2} = k\,2^{j+2}$, level $j+2$

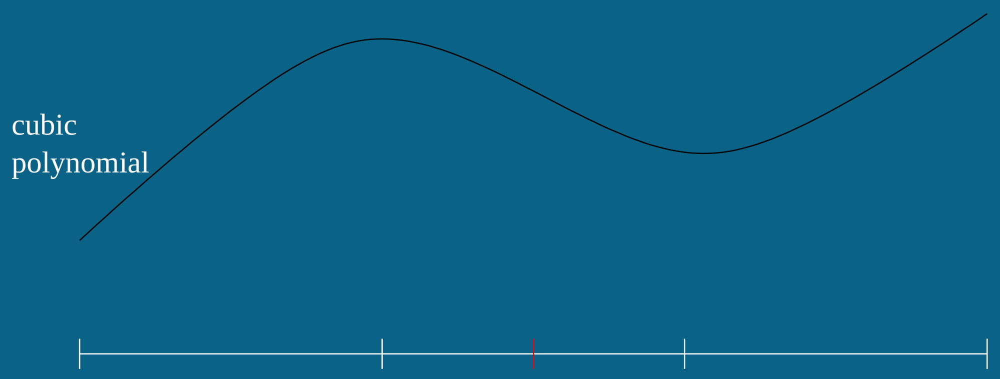
Prediction operator

Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.

back

Prediction operator

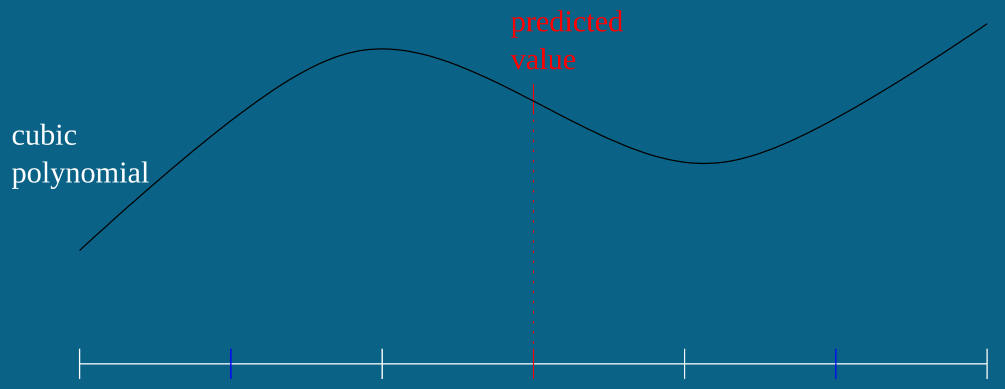
Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.



back

Prediction operator

Predict values at unknown positions of finer level using lagrange interpolating polynomial on coarser level.



$$c^{j+1}_{2k+1} = P_{2N+1}(x^{j+1}_{2k+1}) \text{ and } c^{j+1}_{2k} = c^j_k$$

back

The Algorithm for the Vlasov Problem...

- **Initialisation**: decomposition and compression of f_0.
- **Prediction in** x of the grid $\tilde{G}$ (for important details) at the next split time step following the characteristics forward. Retain points at level just finer.
- **Construction of** $\hat{G}$: grid where we have to compute values of f^* in order to compute its wavelet transform.

back

...The Algorithm for the Vlasov Problem...

- **Advection-interpolation in** x: follow the characteristics backwards in x and interpolate using wavelet decomposition (1): $f^*(x,v) = f^n(x - v\,\Delta t, v)$
- **Wavelet transform of** f^*: compute the c_k and d_k coefficients at the points of $\tilde{G}$.
- **Computation** of electric field from Poisson.
- **Prediction in** v : as for x.

back

...The Algorithm for the Vlasov Problem

- **Construction of $\hat{G}$**: grid where we have to compute values of f^{n+1} in order to compute its wavelet transform.
- **Advection-interpolation in v**: as for x $f^{n+1}(x,v) = f^*(x, v - E(x)\,\Delta t)$ using wavelet decomposition.
- **Wavelet transform of f^{n+1}**: compute the c_k and d_k coefficients at the points of $\tilde{G}$.
- **Compression of f^{n+1}**.

back